IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Robert S. Brayton et al.	§	Group Art Unit: 2178
Serial No.: 10/037,683	§	Examiner: Basehoar, Adam L.
Filed: February 28, 2005	§	
For: Method to Serve Real-Time Data in Embedded Web Server	§	Atty. Docket: 200302369-1 COMP:0270/SWA

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

CERTIFICATE OF MAILING
37 C.F.R. 1.8

I hereby certify that this correspondence is being transmitted by facsimile to the United States Patent and Trademark Office in accordance with 37 C.F.R. § 1.6(d), or is being transmitted via the Office electronic filing system in accordance with 37 C.F.R. § 1.6(a)(4), or is being deposited with the U.S. Postal Service with sufficient postage as First Class Mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on the date below:

September 27, 2007	/Tait R. Swanson/
Date	Tait R. Swanson

APPEAL BRIEF PURSUANT TO 37 C.F.R. §§ 41.31 AND 41.37

This Appeal Brief is being filed in furtherance to the Notice of Appeal and the Pre-Appeal Brief Request for Review electronically filed on May 25, 2007.

In accordance with 37 C.F.R. § 1.136, Appellants request that this and any future reply requiring an extension of time be treated according to the General Authorization For Extensions Of Time previously submitted.

The Commissioner is authorized to charge the requisite fee of $500.00, and any additional fees which may be required, to Deposit Account No. 08-2025; Order No. 200302369-1.

1. REAL PARTY IN INTEREST

The real party in interest is Hewlett-Packard Development Company, L.P., the assignee of the above-referenced application by virtue of the assignment recorded at reel 014628, frame 0103, and dated May 12, 2004.

2. RELATED APPEALS AND INTERFERENCES

Appellants are unaware of any other appeals or interferences related to this Appeal. The undersigned is Appellants' legal representative in this Appeal. Hewlett-Packard Development Company, L.P., the Assignee of the above-referenced application, as evidenced by the documents mentioned above, will be directly affected by the Board's decision in the pending appeal.

3. STATUS OF CLAIMS

Claims 29-48 are currently pending and under final rejection and, thus, are the subject of this Appeal. Claims 1-28 are canceled.

4. STATUS OF AMENDMENTS

The Amendment and Response to Office Action mailed on October 19, 2006 has been entered, and no response to the Final Official Action of April 4, 2007 was filed. Therefore, the present application is not subject to any pending amendments.

5. SUMMARY OF CLAIMED SUBJECT MATTER

This appeal addresses the three independent claims, 29, 37, and 46, and three dependent claims, 34, 45, and 48. Below, Appellants explain each of these claims by identifying specific embodiments in the specification. While these embodiments exemplify the subject matter of the appealed claims, they do not necessarily define the claims' scope. Thus, these claims should not be construed as limited to the following embodiments in virtue of this explanation.

Independent claim 29 recites:

> A method of serving data from a management module of a managed server, comprising:
>
> serving a web page to a requesting computer from a managed server, the web page comprising a source call to an object file, wherein the requesting computer is remote from the managed server;

receiving a request from the requesting computer to the managed server for the object file;

populating the object file in real-time with data from a management module of the managed server after both serving the web page and receiving the request for the object file; and

serving the object file to the requesting computer after populating the object file.

Before addressing the acts recited by claim 29, examples of components that might perform the acts are identified. Examples of the management module and the managed server are identified by reference numbers 106 and 104, respectively, in FIG. 2. An example of the requesting computer is the client 102 of FIG. 2, and examples of the web page are shown in marked-up form by FIGS. 5 and 6 and in code form on page 8. The web page's source call is exemplified by the source tag "scr = file.js" described on page 11 and the source tags "JSFunct.js" and "dataTable.js" on page 8, and an example of the object file called by the source call is exemplified by the JavaScript functions shown on pages 8 and 9. Examples of the data that populates the object file include the current time 424 and the data in frames 402, 404, and 406 of FIG. 6.

The acts recited by claim 29 are exemplified by a dynamic data server process 200 illustrated by FIG. 3. An example of both the act of "serving a web page to a requesting computer from a managed server, the web page comprising a source call to an object file, wherein the requesting computer is remote from the managed server" and the act of "receiving a request from the requesting computer to the managed server for the object file" is the act of transmitting a page/frame from the server to the client, illustrated by block 204. This act is described by the first full paragraph on page 11.

"[P]opulating the object file in real-time with data from a management module of the managed server after both serving the web page and receiving the request for the object file" is exemplified by the act of generating data files for

requested data in real-time at the server, as depicted by block 210. The specification explains this example by stating that the server 104 "generates data files for the requested data in real-time at the server 104 (block 210). For example, the data files may comprise JavaScript functions, variables, arrays, and various other objects for transferring and formatting the dynamic or ephemeral data." Application, page 11.

Finally, the act of "serving the object file to the requesting computer after populating the object file" is exemplified by the act of transmitting data files for the requested data from the server to the client, as illustrated by block 212 of FIG. 3 and described on page 12, lines 1 and 2 of the specification.

Dependent claim 34 recites "The method of claim 29, wherein populating the object file comprises providing a language localization file." An example of a language localization file is described on page 16, lines 9-11 of the application.

Claim 37, the second independent claim, recites:

> A method of displaying a web page, comprising:
>
> requesting at least a frame of a web page from a managed server, wherein the frame comprises a first embedded object;
>
> receiving the frame from the managed server;
>
> requesting data corresponding to the first embedded object from the managed server after receiving the frame from the managed server;
>
> receiving the data corresponding to the first embedded object; and
>
> merging the data corresponding to the first embedded object into the frame.

The subject matter of claim 37 is exemplified by the dynamic data server process 200 illustrated by FIG. 3 and described on page 11, line 4 through page 13, line 3. An example of the act of "requesting at least a frame of a web page from a managed server, wherein the frame comprises a first embedded object" is requesting

a web page from the server, as illustrated by block 202 and described on page 11, lines 5-9. Examples of the embedded object are the JavaScript functions shown on page 8, line 26 through page 9, line 10. The act of "receiving the frame from the managed server" is exemplified by the act of transmitting a page/frame from the server to the client, illustrated by block 204 and described on page 11, lines 8-9. "[R]equesting data corresponding to the first embedded object from the managed server after receiving the frame from the managed server" is exemplified by the act of requesting data from the server to populate the embedded objects, as depicted by block 208 and described on page 11, lines 16-18. The act of "receiving the data corresponding to the first embedded object" is exemplified by the act of transmitting data files for the requested data from the server to the client, as illustrated by block 212 of FIG. 3 and described on page 12, lines 1 and 2 of the specification. Finally, an example of "merging the data corresponding to the first embedded object into the frame" is the act illustrated by block 214 and described on page 12, lines 2-4. An example of the result of merging is shown by the web page 400 of FIG. 6 and described on page 14, line 20 through page 15, line 19.

Dependent claim 45 recites "The method of claim 37, wherein merging the data comprises populating a drop-down menu with a menu item." An example of this act is described on page 12, lines 6-9, stating "the Web browser may merge the unpopulated page or frame(s) with one or more variables or arrays of data for a table, a drop-down menu, a list box, or any other desired application or objected-oriented feature within the page or frame(s)." An example of a drop-down menu is illustrated by the drop-down menus 408 and 410 of FIG. 5 and described on page 15, lines 2-3, and examples of menu items are illustrated by menu items 426 and 428 of FIG. 6 and described on page 15, lines 14-15.

Claim 46, the third independent claim subject to the present appeal, recites:

A server, comprising:

a management module configured to generate dynamic data; and

> a file system storing a web page that has both a first embedded object configured to access the dynamic data and a second embedded object configured to merge the dynamic data with the web page, wherein the first embedded object is executable on a client remote from the server to request the dynamic data.

The subject matter of claim 46 is exemplified by the dynamic data server system 100 of FIG. 2, described on page 7, line 15 through page 11, line 2. An example of the management module is identified by reference number 106 and is described on page 7, line 20 through page 8, line 4. Examples of the dynamic data include the time at which the dynamic data is gathered 424 (FIG. 6 and page 15, lines 6-9) and the dynamic data 116 generated by the management module 106 of the server 104 (FIG. 2 and page 7, line 4 through page 9, line 21). The file system is exemplified by the file system 112 of FIG. 2 (page 7, lines 20-21), and examples of the web page are shown in marked-up form by FIGS. 5 and 6 (page 14, line 20 though page 15, line 19) and in code form on page 8, lines 9-21. An example of the first embedded object is the JavaScript function titled *dataTable.js* shown on page 9, lines 6-10, and an example of the second embedded object is the JavaScript function titled *JSFunct.js* shown on the same page 8, line 26 through page 9, line 3.

Dependent claim 48 recites "The server of claim 46, comprising a lights-out management module." An example of a lights-out management module is illustrated by the LOM (lights-out management module) 108 of FIG. 2 and described on page 7, line 23 through page 8, line 2, which recites "The lights out management module 108 is provided for managing a headless computer system, which generally represents a computer lacking user interaction devices, such as a monitor, a keyboard, and a mouse."

6. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Appellants respectfully urge the Board to reverse the following rejections:

1) claims 29-33, 35-44 and 46-47 as obvious over Pettersen (U.S. Patent No. 6,826,594, hereinafter "the Pettersen reference") in

view of Chen et al. (U.S. Patent No. 6,021,437, hereinafter "the Chen reference");

2) claim 34 as obvious over Pettersen in view of Chen and Thurston (U.S. Patent No. 6,865,716);
3) claim 45 as obvious over Pettersen in view of Chen and Lynch et al. (U.S. Patent No. 6,823,319, hereinafter "the Lynch reference"); and
4) claim 48 as obvious over Pettersen in view of Chen and Ellison et al. (U.S. Patent No. 6,487,547, hereinafter "the Ellison reference").

7. ARGUMENT

To reject the pending claims, the Examiner misapplied long-standing and binding legal precedent. Below, Appellants summarize this precedent and explain the manner in which is has been misapplied. Appellants explain that the rejections are flawed because Pettersen cannot be properly combined with Chen and because even if hypothetically combined, these references do not teach all of the features of the present claims. Faced with these flaws, the Examiner should have allowed this application in view of the response that Appellants filed on January 19, 2006. Because the Examiner failed to do this, Appellants turn to the Board and respectfully request full and favorable consideration of claims 29-48.

Legal Precedent

The pending claims must be given an interpretation that is reasonable and consistent with the *specification*. *See In re Prater*, 415 F.2d 1393, 1404-05, 162 U.S.P.Q. 541, 550-51 (C.C.P.A. 1969) (emphasis added); *see also In re Morris,* 127 F.3d 1048, 1054-55, 44 U.S.P.Q.2d 1023, 1027-28 (Fed. Cir. 1997); *see also* M.P.E.P. §§ 608.01(o) and 2111. Indeed, the specification is "the primary basis for construing the claims." *See Phillips v. AWH Corp.*, No. 03-1269, -1286, at 13-16 (Fed. Cir. July 12, 2005) (*en banc*). One should rely *heavily* on the written description for guidance as to the meaning of the claims. *See id.*

Interpretation of the claims must also be consistent with the interpretation that one of ordinary skill in the art would reach. *See In re Cortright*, 165 F.3d 1353, 1359, 49 U.S.P.Q.2d 1464, 1468 (Fed. Cir. 1999); M.P.E.P. § 2111. "The inquiry into how a person of ordinary skill in the art understands a claim term provides an objective baseline from which to begin claim interpretation." *See Collegenet, Inc. v. ApplyYourself, Inc.*, 418 F.3d 1225, 75 U.S.P.Q.2d 1733, 1738 (Fed. Cir. 2005) (quoting *Phillips v. AWH Corp.*, 75 U.S.P.Q.2d 1321, 1326). The Federal Circuit has made clear that derivation of a claim term must be based on "usage in the ordinary and accustomed meaning of the words amongst artisans of ordinary skill in the relevant art." *See id.*

The burden of establishing a *prima facie* case of obviousness falls on the Examiner. *Ex parte Wolters and Kuypers*, 214 U.S.P.Q. 735 (PTO Bd. App. 1979). In addressing obviousness determinations under 35 U.S.C. § 103, the Supreme Court in *KSR International Co. v. Teleflex Inc.*, No. 04-1350 (April 30, 2007), reaffirmed many of its precedents relating to obviousness including its holding in *Graham v. John Deere Co.,* 383 U.S. 1 (1966). In *Graham*, the Court set out an objective analysis for applying the statutory language of §103:

> Under §103, the scope and content of the prior art are to be determined, differences between the prior art and the claims at issue are to be ascertained, and the level of ordinary skill in the pertinent art are to be resolved. Against this background the obviousness or non-obviousness of the subject matter is to be determined. Such secondary considerations as commercial success, long-felt but unresolved needs, failure of others, etc., might be utilized to give light to the circumstances surrounding the origin of the subject matter sought to be patented. *KSR*, s*lip op.* at 2 (citing *Graham*, 383 U.S. at 17-18).

In *KSR*, the Court also reaffirmed that "a patent composed of several elements is not proved obvious merely by demonstrating that each of its elements was, independently, known in the prior art." *Id.* at 14. In this regard, the *KSR* court stated that "it can be important to identify a reason that would have prompted a

person of ordinary skill in the relevant field to combine the elements in the way the claimed new invention does ... because inventions in most, if not all, instances rely upon building blocks long since uncovered, and claimed discoveries almost of necessity will be combinations of what, in some sense, is already known." *Id.* at 14-15. Traditionally, to establish a *prima facie* case of obviousness, the CCPA and the Federal Circuit have required that the prior art not only include all of the claimed elements, but also some teaching, suggestion, or motivation to combine the known elements in the same manner set forth in the claim at issue. *See, e.g., ASC Hospital Systems Inc. v. Montifiore Hospital,* 221 U.S.P.Q. 929, 933 (Fed. Cir. 1984) (holding that obviousness cannot be established by combining the teachings of the prior art to produce the claimed invention absent some teaching or suggestion supporting the combination.); *In re Mills,* 16 U.S.P.Q.2d 1430, 1433 (Fed. Cir. 1990) (holding that the mere fact that references can be combined or modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the combination). In *KSR*, the court noted that the demonstration of a teaching, suggestion, or motivation to combine provides a "helpful insight" in determining whether claimed subject matter is obvious. *KSR, slip op.* at 14. However, the court rejected a *rigid* application of the "TSM" test. *Id.* at 11. In this regard, the court stated:

> The obviousness analysis cannot be confined by a formalistic conception of the words teaching, suggestion, and motivation, or by overemphasis on the importance of published articles and explicit content of issued patents. The diversity of inventive pursuit and of modern technology counsels against limiting the analysis in this way. In many fields it may be that there is little discussion of obvious techniques or combinations, and it often may be the case that market demand, rather than scientific literature, will drive design trends. *Id.* at 15.

In other words, the *KSR* court rejected a rigid application of the TSM test which requires that a teaching, suggestion or motivation to combine elements in a particular manner must be explicitly found in the cited prior art. Instead, the *KSR*

court favored a more expansive view of the sources of evidence that may be considered in determining an apparent reason to combine known elements by stating:

> Often, it will be necessary for a court to look to interrelated teachings of multiple patents; the effects of demands known to the design community or present in the marketplace; and the background knowledge possessed by a person having ordinary skill in the art all in order to determine whether there was an apparent reason to combine in the known elements in the fashion claimed in the patent at issue. *Id.* at 14.

The *KSR* court also noted that there is not necessarily an inconsistency between the idea underlying the TSM test and the *Graham* analysis, and it further stated that the broader application of the TSM test found in certain Federal Circuit decisions appears to be consistent with *Graham*. *Id.* at 17-18 (citing *DyStar Textilfarben GmbH and Co. v. C.H. Patrick Co.,* 464 F.3d 1356, 1367 (2006) ("Our suggestion test is in actuality quite flexible and not only permits but *requires* consideration of common knowledge and common sense"); *Alza Corp. v. Mylan Labs, Inc.,* 464 F.3d 1286, 1291 (2006) ("There is flexibility in our obviousness jurisprudence because a motivation may be found *implicitly* in the prior art. We do not have a rigid test that requires a teaching to combine ... ")).

Furthermore, the *KSR* court did not diminish the requirement for objective evidence of obviousness. *Id.* at 14 ("To facilitate review, this analysis should be made explict. See *In re Kahn,* 441 F.3d 977, 988 (CA Fed. 2006) ("[R]ejections on obviousness grounds cannot be sustained by mere conclusory statements; instead, there must be some articulated reasoning with some rational underpinning to support the legal conclusion of obviousness"). As our precedents make clear, however, the analysis need not seek out precise teachings directed to the specific subject matter of the challenged claim, for a court can take account of the inferences and creative steps that a person of ordinary skill in the art would employ."); *see also, In re Lee,* 61 U.S.P.Q.2d 1430, 1436 (Fed. Cir. 2002) (holding that the factual inquiry whether

to combine references must be thorough and searching, and that it must be based on *objective evidence of record*).

When prior art references require a selected combination to render obvious a subsequent invention, there must be some reason for the combination other than the hindsight gained from the invention itself, i.e., something in the prior art as a whole must suggest the desirability, and thus the obviousness, of making the combination. *Uniroyal Inc. v. Rudkin-Wiley Corp.*, 837 F.2d 1044, 5 U.S.P.Q.2d 1434 (Fed. Cir. 1988). One cannot use hindsight reconstruction to pick and choose among isolated disclosures in the prior art to deprecate the claimed invention. *In re Fine*, 837 F.2d 1071, 5 U.S.P.Q.2d 1596 (Fed. Cir. 1988). The Federal Circuit has warned that the Examiner must not, "fall victim to the insidious effect of a hindsight syndrome wherein that which only the inventor taught is used against its teacher." *In re Dembiczak*, F.3d 994, 999, 50 U.S.P.Q.2d 52 (Fed. Cir. 1999) (quoting *W.L. Gore & Assoc., Inc. v. Garlock, Inc.*, 721 F.2d 1540, 1553, 220 U.S.P.Q. 303, 313 (Fed. Cir. 1983)).

It is improper to combine references where the references teach away from their combination. *In re Grasselli*, 713 F.2d 731, 743, 218 U.S.P.Q. 769, 779 (Fed. Cir. 1983); M.P.E.P. § 2145. Moreover, if the proposed modification or combination of the prior art would change the principle of operation of the prior art invention being modified, then the teachings of the references are not sufficient to render the claims *prima facie* obvious. *In re Ratti*, 270 F.2d 810, 123 U.S.P.Q. 349 (CCPA 1959); *see* M.P.E.P. § 2143.01(VI). If the proposed modification or combination would render the prior art invention being modified unsatisfactory for its intended purpose, then there is no suggestion or motivation to make the proposed modification. *In re Gordon*, 733 F.2d 900, 221 U.S.P.Q. 1125 (Fed. Cir. 1984); *see* M.P.E.P. § 2143.01(V).

First Rejection

In the Office Action, the Examiner rejected claims 29-33, 35-44 and 46-47 as obvious over the Pettersen reference in view of the Chen reference. Appellants respectfully traverse this rejection.

Improper Combination - Lack of Objective Evidence of Reasons to Modify/Combine

The Examiner has not shown the requisite motivation or suggestion to modify or combine the cited references to reach the present claims. The Examiner must provide objective evidence, rather than subjective belief and unknown authority, of the requisite motivation or suggestion to combine or modify the cited references. *In re Lee*, 61 U.S.P.Q.2d. 1430 (Fed. Cir. 2002). In the present rejection, the Examiner combined the Pettersen and Chen references based on the conclusory and subjective statement that it would have been obvious "for the dynamic data of Pettersen to have been created in real-time as disclosed in Chen et al, because Chen et al teach said process provides a simple, effective, and inexpensive to implement way for real-time monitoring of data column 2, lines 31-38)." Final Office Action Mailed on May 19, 2006, page 3. However, this is merely the speculative opinion of the Examiner that is inconsistent with Chen, who already teaches a technique for constructing a web page with the server data, and Examiner has not provided any evidence to indicate why one of ordinary skill in the art would find the technique taught by Chen inadequate and replace it with the process for forming a web page taught by Pettersen.

In the Office Action mailed October 19, 2006, the Examiner attempted to address this deficiency, remarking:

> Petterson teaches a method for delivering dynamic content from a server to a client browser after a web page has been provided to the client browser. The Petterson reference also teaches wherein the dynamic content was generated at runtime. The Petterson reference however does not specifically teach wherein

> the data was indicative of a real-time current status of a managed server. Chen et al. cures this deficiency by teaching creating dynamic data indicative of the status of a managed server generated in real-time and delivering said dynamic data to a client browser (Abstract; column 2, lines 31-67, column 4, lines 10-19). Chen et al. also teach that creating dynamic data in real-time provides the benefit of a simple, effective, and inexpensive way to implement real-time monitoring of data (column 2, lines 31-38). Thus Chen et al. provides the Petterson reference the ability for a user to provide administration and maintenance support of a data processing system.
>
> Office Action mailed October 19, 2006, page 12.

However, the Examiner still has not provided a valid reason to combine Chen with Pettersen. One of skill in the art would not combine these references unless the combination provided a benefit that neither reference provided in isolation. That is, the combination must have some utility over each reference; otherwise, there would have been no reason to combine them. To do otherwise adds complexity and cost with no benefit. In the present case, if Chen's system accomplishes a given task, there would have been no reason to combine Chen with Pettersen to accomplish the same task, yet the Examiner speculates that an artisan would combine Chen with Pettersen because the combination would offer "real-time monitoring of data" and "the ability for a user to provide administration and maintenance support of a data processing system." However, the system taught by Chen already offered these benefits. Thus, an artisan would simply have implemented the system taught by Chen to achieve "real-time monitoring of data" and "the ability for a user to provide administration and maintenance support of a data processing system" without having had any motivation to turn to the Petterson reference. The Examiner still has not met his burden to explain why one of ordinary skill in the art would combine these references.

The References Teach Away from the Proposed Combination

As summarized above, it is improper to combine references where the references teach away from their combination. *In re Grasselli*, 713 F.2d 731, 743, 218 U.S.P.Q. 769, 779 (Fed. Cir. 1983); M.P.E.P. § 2145. Here, the Petterson reference teaches away from acquiring data in the manner taught by the Chen reference.

As described in more detail below, the Pettersen reference teaches a remote content management system in which a web site and advertisements to be inserted into the web site are requested from different servers. *See* Pettersen, col. 6, ll. 8-21, and Fig. 11. The primary web page 791 comes from the affiliate web server 791, and the inserted advertisement data comes from the server 781 for a content serving website 780. *Infra.* To explain the utility of such a system, the Petterson reference emphasizes both the importance of storing the advertisement data on a remote server and the problems associated with placing the advertisement data on the affiliate's web server. *See* Pettersen, col. 2, ll. 32-40. Thus, the Petterson reference teaches away from acquiring the primary web page and the data inserted into the web page from the same server.

In contrast, Chen teaches precisely the arrangement of web servers and inserted data that the Pettersen system is designed to avoid. The Chen reference teaches intelligent agents that fully construct a web page on a server before serving the web page to a client. *See* Chen, col. 3, ll. 31-32. In Chen, the real-time server data comes from the same server that provides the primary web page in which the real-time server data is inserted. Thus, the Chen reference teaches web pages that are received by the client in their entirety, so Chen teaches acquiring the web page and the inserted data from the same server.

The proposed modification is improper because the references teach away from one another. The Petterson reference teaches that frequently changing data that is inserted into a web page should not be acquired from the same server as the web

page, and the Chen reference teaches acquiring the real-time server data from the same server that provides the web page in which the data is inserted. Because these references teach away from each other, one of ordinary skill in the art would not combine the cited references as proposed. For this reason also, the Examiner has not met his burden.

Features of Claim 29 that are Missing from the Cited References

The Examiner's rejection is flawed for a number of other reasons, not the least of which being that the cited references do not teach all of the recited claim features. In claim 29, the same server both serves a web page and receives a request for an object file. Claim 29 recites "serving a web page to a requesting computer from a managed server," and "receiving a request from the requesting computer to the managed server for the object file." (Emphasis added.) Both of these phrases refer to the same managed server. The cited references, however, do not teach or suggest, alone or in hypothetical combination, a managed server that both serves a web page comprising a source call to an object file and receives a request for the object file.

The Pettersen reference teaches a remote content management system in which a web site and advertisements to be inserted into the web site are requested from different servers. *See* Pettersen, col. 6, ll. 8-21, and Fig. 11. Specifically, the Pettersen reference teaches serving a website with the following steps:

1) an affiliate web server 791 serves a web page 793 to a user system 760 (Pettersen, Fig. 11 and column 6, lines 39-41);
2) the user system 760 then requests advertisements from a web server 781 for a content serving website 780 (Pettersen, Fig. 11 and column 7, lines 13-16); and
3) the user system 760 then inserts the advertisements into the web page 793 (Pettersen, Fig. 11 and column 7, lines 34-37).

Pettersen's affiliate web server 791 is not the same server as the web server 781, so the request to the web server 781 for advertisements is not a request to the affiliate web server 791. In other words, the user system 760 does not request the advertisements from the server 791 that provides the web site. *See id* at col. 7, ll. 5-15. That is, different servers provide the web site and the inserted advertisement data. Thus, the Petterson reference does not teach or suggest a managed server that both serves a web page comprising a source call to an object file and receives a request for the object file.

The Chen reference does not teach or suggest this feature, either. Indeed, the Chen reference does not even teach an object file, let alone a managed server that receives a request for an object file. The Chen reference teaches intelligent agents that fully construct a web page on a server before serving the web page to a client. *See* Chen, col. 3, ll. 31-32. Thus, because the web page is received by the client in its entirety, these web pages do not prompt a client computer to request additional data, such as an object file, after receipt of the web pages. Accordingly, the Chen reference does not teach receiving a request to a managed server for an object file.

In view of these deficiencies, among others, the Pettersen reference and the Chen reference, taken alone or in hypothetical combination, cannot render obvious claim 29 or the claims that depend therefrom.

Features of Claim 37 that are Missing from the Cited References

The Examiner's rejection of claim 37 is also flawed. The cited references do not teach or suggest, alone or in hypothetical combination, "requesting data corresponding to the first embedded object from the managed server after receiving the frame from the managed server," as recited by independent claim 37. (Emphasis added.) That is, in claim 37, data corresponding to the first embedded object is requested from the same server that provided the frame. In contrast, as discussed above, the Pettersen reference teaches a remote content management system in which a web site and advertisements to be inserted into the web site are requested

from different servers. *See* Pettersen, col. 6, ll. 8-21, and Fig. 11. Thus, even if, *ad arguendo*, the web server 791 for the affiliate site 190 were a managed server, the advertisements would be provided by another web server 781 associated with the advertisements, and any requests for the advertisements would not be from the server 791 that provides the web site. *See id.* at col. 7, ll. 5-15. Further, as noted above, the Chen reference teaches web pages that are received by the client in their entirety. In other words, these web pages do not prompt a client computer to request additional data from the server providing the web pages, so Chen does not teach requesting data corresponding to an embedded object from any server, let alone from a server that previously provided a frame. Thus, the cited references, taken alone or in hypothetical combination, do not teach or suggest all the steps of independent claim 37.

In an attempt to address these flaws, the Examiner asserted the following:

> In regard to independent claims 29 and 37, Applicant argues that the Pettersen reference fails to teach or suggest where the managed server both serves the web page with the source call and receives a request for the object file. The Examiner respectfully disagrees and believes that Pettersen teaches that one embodiment was a client and single server system wherein the client could be the affiliate web site and the host server sending both the web page and the object file could be the content serving web site (column 8, lines 43-67; column 9, lines 1-31). Additionally, the Examiner believes that the preferred embodiment of the Pettersen reference (i.e. the two server system) would have been obvious to modify in view of the single managed server of the Chen reference. In this case, Pettersen generally teaches a method for delivering dynamic content from a server to a client browser after a web page had been provided to the client browser. The Pettersen reference also teaches wherein the dynamic content was generated at run time. The Pettersen reference however does not specifically teach wherein the data that was to be included in said web page was indicative of a real-time current status of a managed server. Chen et al cures this deficiency by teaching creating dynamic data

> indicative of the status of a managed server generated in real-time and delivering a web page with said dynamic data to a client browser (Abstract; column 2, lines 31-67, column 4, lines 10- 19). Chen et al also teach that creating dynamic data in real-time provides the benefit of a simple, effective, and inexpensive way to implement real-time monitoring of data (colunm 2, lines 31-38). Thus Chen et al would have provided the Pettersen reference the ability for user system to provide administration and maintenance support of the affiliate web sites servers. Final Office Action mailed April 4, 2007, pages 11 and 12.

In this passage, the Examiner makes two arguments: 1) that Pettersen teaches a single-server embodiment that serves both the web page and dynamic data, and 2) that Pettersen can be modified in light of Chen to reach the subject matter of claims 29 and 37. *See id.* Both of these arguments are flawed.

Contrary to the first argument, Pettersen does not teach a single-server embodiment. Indeed, as explained above in the section titled *The References Teach Away from the Proposed Combination*, the Pettersen reference teaches away from such an embodiment. Pettersen emphasizes both the importance of storing the advertisement data on a remote server and the problems associated with placing the advertisement data on the affiliate's web server. *See* Pettersen, col. 2, ll. 32-40. That is, the Pettersen reference teaches away from acquiring the primary web page and the data inserted into the web page from the same server.

To support the first argument, that Pettersen teaches a single-server embodiment, the Examiner cited column 8, line 43-column 9, line 31 of Pettersen (hereinafter "the cited passage"). Final Office Action mailed April 4, 2007, page 11. The cited passage, however, does not disclose or suggest the claimed managed server, which as explained above, both serves a web page comprising a source call to an object file and receives a request for the object file. The cited passage merely discusses the relationship between a computer 80 at an affiliate web site 790 and a server 90 at a content serving web site 70. Pettersen, column 8, lines 47-50. In describing this relationship, the cited passage does not state that the server 90 serves

the internal web page 86; it states that the server 90 serves an advertisement (i.e., retrieved output 98) for insertion into the internal web page 86. *See* Pettersen, column 8, lines 61-63. As suggested by the word "internal" and explained elsewhere in Pettersen, though not explicitly stated in the cited passage, the internal web page 86 is local to the client computer 80 at the affiliate website 790. *See* Pettersen, column 10, lines 20-22. The cited passage does not support the Examiner's contention that Pettersen teaches a single server that both serves a web page comprising a source call to an object file and receives a request for the object file. Thus, the Examiner's first argument is wrong.

The Examiner's second argument, that it would have been obvious to modify Pettersen in view of Chen, is also flawed for reasons that are explained above. Specifically, in the section titled *Improper Combination - Lack of Objective Evidence of Reasons to Modify/Combine*, Appellants explain that the Examiner has failed to identify a convincing rationale why it would have been obvious to combine Pettersen with Chen, and in the section titled *The References Teach Away from the Proposed Combination*, Appellants explain that Pettersen teaches away from such a combination. Notably, in the Final Office Action, the Examiner completely failed to address either of these arguments, choosing instead to merely reiterate the same conclusory obviousness analysis from the previous office action. Mere speculation by the Examiner is inadequate – the rationale must be supported by objected evidence.

Features of Claim 46 that are Missing from the Cited References

Independent claim 46 recites a "first embedded object [that] is executable on a client remote from the server to request the dynamic data." (Emphasis added.) In contrast, the Pettersen reference teaches retrieving static, pre-stored advertisements and inserting the static advertisements in a web page dynamically. *See e.g.*, Pettersen, col. 7, ll. 11-13; col. 8, ll. 37-42; and col. 10, ll. 35-46. The advertisements are stored in a smart zone content database 785 (*Id.* at col. 7, ll. 11-13) and are dynamically inserted into a web page (*See id.* at col. 7, ll. 6-8) when the web page is displayed. However, the advertisements do not change, i.e., the

advertisement data is static. *See id.* at col. 7, ll. 52-59. Indeed, the disclosed method of changing the advertisements in the database 785 is manually editing the list of advertisements. *See id.* Thus, the Pettersen reference does not teach or suggest an embedded object that is executable to request dynamic data. Further, as noted above, both the Chen reference and the Thurston reference teach web pages that do not prompt a client computer to request additional data. Therefore, because these web pages do not request additional data, they clearly do not include embedded objects that are executable on a client to request dynamic data. Accordingly, the cited references, taken alone or in hypothetical combination, do not teach or suggest all of the features recited by independent claim 46.

In the Final Office Action, the Examiner attempted to address this inadequacy, remarking as follows:

> In regard to independent claims 29 and 46, Applicant argues that neither of the references teach "a first embedded object is executable on a client remote from the server to request the dynamic data". As claimed, The Examiner does not find the Applicant's arguments persuasive in regards to attempting to differentiate the dynamic content of the Pettersen reference and the dynamic data of the claimed invention. In general dynamic data has been given its broadest reasonable interpretation as content that exists in a database on a web server that is pulled or placed on the page a user is viewing a the time it is needed rather than at a predetermined or fixed time. Pettersen clearly teaches wherein the dynamic data that could be inserted into the requested web page could be image based content, regular hyper-linked text....and the like (column 4, lines 10-17; column 6, lines 24-25; column 7, lines 6-43). Because Pettersen further teaches wherein the requested content could be modified by the affiliate web site (column 7, lines 1-5) the content therefore has not been considered as just static data. Pettersen also teaches wherein the look and behavior of the dynamic content could be varied according to a content modification level and could include run-time modified content (Abstract; column

10, lines 6-22: "revenue link....dynamically updated").
Final Office Action mailed April 4, 2007, page 12.

In this argument, the Examiner has defined "dynamic data" as "content that exists in a database on a web server that is pulled or placed on the page a user is viewing at the time it is needed rather than at a predetermined or fixed time," and the Examiner has attempted to apply this definition to three instances where Pettersen mentions data: the description of the pre-stored advertisements that are inserted in a web page; the discussion of the affiliate website 790 modifying the advertisements stored at the content-serving website 780; and the "run-time modified content" mentioned by the abstract. None of these instances, however, teach or suggest "dynamic data."

One of ordinary skill in the art would not arrive at the Examiner's construction of "dynamic data." The Examiner has construed the term "dynamic data" so broadly that he deprives other terms in the present claim of meaning. The claim recites "a web page that has ... a first embedded object configured to access the dynamic data." The Examiner's definition, however, encompasses the web page itself, as web page data typically exists in a web server and is delivered at the time a web page is served, i.e., as needed. Thus, the Examiner's construction equates "dynamic data" with "web page." One of ordinary skill in the art, however, would understand the term "dynamic data" to describe something other than simply a web page.

A plausible construction of "dynamic data" would account for the term's context in the claim. The claim recites "a management module configured to generate dynamic data," indicating that the "dynamic data" is generated rather than merely recalled. Further, the claim recites "a first embedded object configured to access the dynamic data and a second embedded object configured to merge the dynamic data." This indicates that the adjective "dynamic" modifies the noun "data" to have some property other than just being accessed and merged, otherwise both the first embedded object and the second embedded object are superfluous.

That is, if "dynamic" merely means accessed and merged with a web page, there would be no reason to both label the data as dynamic and recite the first embedded object and the second embedded object that perform these functions. Thus, the term "dynamic data" does not encompass Pettersen's stored advertisements that are accessed and inserted into a web page.

Although Appellants do not suggest that the specification be read into the claims, the Appellants stress that the specification provides further context for the term "dynamic data." On page 8, the specification states "The system 100 serves dynamic data, or real-time data, to the client 102." The phrase "or real-time data" is set off by commas, indicating that it is a non-restrictive clause, i.e., that it re-names the preceding term—dynamic data. In another passage, the specification states "The technique constructs the object files in the embedded system in real-time separate from the Web pages. The technique then merges the dynamic data with the Web pages and displays the data-populated Web pages via a Web browser." Application, page 5. The phrase "the dynamic data" in this passage uses the definite article "the" to refer back to the real-time data. The object files constructed in real time serve as the antecedent for the phrase "the dynamic data." Thus, the specification describes dynamic data as real-time data, or data created in real time.

In contrast, as explained above, the Pettersen reference teaches retrieving static, pre-stored advertisements and inserting the static advertisements in a web page dynamically. *See e.g.*, Pettersen, col. 7, ll. 11-13; col. 8, ll. 37-42; and col. 10, ll. 35-46. Thus, the Pettersen reference does not teach or suggest an embedded object that is executable to request dynamic data; it teaches dynamically inserting static data. Further, the Chen and Thurston references teach web pages that do not prompt a client computer to request additional data. Therefore, because these web pages do not request additional data, they clearly do not include embedded objects that are executable on a client to request dynamic data. Accordingly, the cited references, taken alone or in hypothetical combination, do not teach or suggest all of the features recited by independent claim 46.

Second Rejection

In the Office Action, the Examiner rejected claim 34 as obvious over the Pettersen reference in view of the Chen reference and the Thurston reference. Appellants respectfully traverse this rejection.

The rejection of claim 34, which depends from claim 29, is defective for at least the reasons set forth above with respect to the rejection of claim 29 under Section 103. Specifically, neither the Chen reference nor the Pettersen reference, alone or in hypothetical combination, teach or suggest a managed server that both serves a web page comprising a source call to an object file and receives a request for the object file, as recited by parent claim 29, and the Thurston reference does not cure this deficiency. Accordingly, the Pettersen reference, the Chen reference and the Thurston reference, taken alone or in hypothetical combination, cannot render claim 34 obvious.

Third Rejection

In the Office Action, the Examiner rejected claim 45 as obvious over the Pettersen reference in view of the Chen reference and the Lynch reference. Appellants respectfully traverse this rejection.

The rejection of claim 45, which depends from claim 37, is defective for at least the reasons set forth above with respect to the rejection of claim 37 under Section 103. Specifically, neither the Chen reference nor the Pettersen reference, alone or in hypothetical combination, teach or suggest "requesting data corresponding to the first embedded object from the managed server after receiving the frame from the managed server," as recited by parent claim 37 (emphasis added), and the Lynch reference does not cure this deficiency. Accordingly, the Pettersen reference, the Chen reference and the Lynch reference, taken alone or in hypothetical combination, cannot render claim 45 obvious.

Fourth Rejection

In the Office Action, the Examiner rejected claim 48 as obvious over the Pettersen reference in view of the Chen reference and the Ellison reference. Appellants respectfully traverse this rejection.

The rejection of claim 48, which depends from claim 46, is defective for at least the reasons set forth above with respect to the rejection of claim 46 under Section 103. Specifically, neither the Chen reference nor the Pettersen reference, alone or in hypothetical combination, teach or suggest a "first embedded object [that] is executable on a client remote from the server to request the dynamic data," as recited by parent claim 46 (emphasis added), and the Ellison reference does not cure this deficiency. Accordingly, the Pettersen reference, the Chen reference and the Ellison reference, taken alone or in hypothetical combination, cannot render claim 48 obvious.

For these reasons among others, Appellants respectfully request the Board to reverse each of the rejections under 35 U.S.C. § 103.

Conclusion

In view of the above remarks, Appellants respectfully submit that the Examiner has provided no supportable position or evidence that claims 29-48 are obvious under Section 103. Accordingly, Appellants respectfully request that the Board find claims 29-48 patentable over the prior art of record, reverse all outstanding rejections, and allow claims 29-48.

Respectfully submitted,

Date: September 27, 2007

/Tait R. Swanson/
Tait R. Swanson
Registration No. 48,226
(281) 970-4545

HEWLETT-PACKARD COMPANY
Intellectual Property Administration
P.O. Box 272400
Fort Collins, Colorado 80527-2400

8. APPENDIX OF CLAIMS ON APPEAL

29. (previously presented) A method of serving data from a management module of a managed server, comprising:

serving a web page to a requesting computer from a managed server, the web page comprising a source call to an object file, wherein the requesting computer is remote from the managed server;

receiving a request from the requesting computer to the managed server for the object file;

populating the object file in real-time with data from a management module of the managed server after both serving the web page and receiving the request for the object file; and

serving the object file to the requesting computer after populating the object file.

30. (previously presented) The method of claim 29, wherein populating the object file comprises populating the object file with a scripting function.

31. (previously presented) The method of claim 30, wherein the scripting function is a JavaScript function.

32. (previously presented) The method of claim 29, wherein populating the object file comprises populating the object file with an array of data.

33. (previously presented) The method of claim 29, wherein populating the object file comprises acquiring real-time data indicative of a current status of a server.

34. (previously presented) The method of claim 29, wherein populating the object file comprises providing a language localization file.

35. (previously presented) The method of claim 29, wherein serving the web page comprises serving a web page configured for a handheld or palmtop computing platform.

36. (previously presented) The method of claim 29, wherein serving a web page comprises serving a web page across a firewall.

37. (previously presented) A method of displaying a web page, comprising:

requesting at least a frame of a web page from a managed server, wherein the frame comprises a first embedded object;

receiving the frame from the managed server;

requesting data corresponding to the first embedded object from the managed server after receiving the frame from the managed server;

receiving the data corresponding to the first embedded object; and

merging the data corresponding to the first embedded object into the frame.

38. (previously presented) The method of claim 37, comprising displaying the frame.

39. (previously presented) The method of claim 37, comprising evaluating the frame to identify a source tag of the embedded object.

40. (previously presented) The method of claim 37, wherein the data corresponding to the embedded object comprises dynamic data from a management module of the managed server.

41. (previously presented) The method of claim 40, wherein the dynamic data is generated in real-time in response to the request for data corresponding to the embedded object.

42. (previously presented) The method of claim 37, wherein the data corresponding to the first embedded object comprises a scripting language function.

43. (previously presented) The method of claim 42, wherein the frame comprises a second embedded object linked to dynamic data in the managed server, and wherein the scripting language function is configured to merge the dynamic data with the frame.

44. (previously presented) The method of claim 37, wherein the data corresponding to the first embedded object comprises the current time and dynamic data gathered at the managed server at the current time.

45. (previously presented) The method of claim 37, wherein merging the data comprises populating a drop-down menu with a menu item.

46. (previously presented) A server, comprising:

a management module configured to generate dynamic data; and

a file system storing a web page that has both a first embedded object configured to access the dynamic data and a second embedded object configured to merge the dynamic data with the web page, wherein the first embedded object is executable on a client remote from the server to request the dynamic data.

47. (previously presented) The server of claim 46, wherein the second embedded object is executable on a client remote from the server to merge the dynamic data with the web page.

48. (previously presented) The server of claim 46, comprising a lights-out management module.

9. EVIDENCE APPENDIX

none

10. RELATED PROCEEDINGS APPENDIX

none